

Resume Maintaining System for Referral Using Cloud Computing

Pranjali A. Pali ¹, N.D. Kale ²P.G. Student, Department of Computer Engineering, TSSM'S PVPIT, Bavdhan, Pune, Maharashtra India¹Associate Professor, Department of Computer Engineering, TSSM'S PVPIT, Bavdhan, Pune, Maharashtra India²

ABSTRACT: As Cloud Computing becomes prevalent, more and more sensitive information are being centralized into the cloud, such as emails, personal health records, private videos and photos, finance data, government documents, etc. By storing data into the cloud, the data owners can be relieved from the burden of data storage and maintain it, so as to enjoy the on-demand high quality data storage service. Cloud computing provides many benefits considering low cost and accessibility of data. Ensuring the security of cloud computing is a major factor in the cloud computing environment as user often store sensitive information or data with cloud storage providers but these providers may be untrusted. In this paper, We introduce system providing fault-tolerant storage over multiple cloud storage providers. The cloud computing use has rapidly increased, security in cloud computing still considered a major issue in the cloud computing environment. User do not want to lose their private information as a result of malicious attack in the cloud, related to the service the loss of service availability has caused many problems for a large number of user. Therefore data intrusion leads to many problems for users of cloud computing. We expect to develop a Resume Maintaining System which will offer a service based on cloud computing concept using window share point technology. It will provide the facility to store the user details with educational and work experience details along with Resume file Storing facility in the cloud via SharePoint application and will allow user to access them. Service application will allow user to upload their Resume File to the cloud, Via SharePoint Application And will provide an automated facility of checking the uploaded Resume File links. If the system finds that the link is broken then the automated report will be generated for the broken links and the application owner will see the report on his dashboard administrator then can request the users to upload the new file to establish the broken links back.

KEYWORDS: Cloud Computing, fault tolerance, recovery, implementation, Availability, Data Privacy

I. INTRODUCTION

Cloud Computing is the long dreamed vision of computing as a utility, where users can store their data into the cloud so as to enjoy the on demand high quality applications and services from a shared pool of configurable computing resources. The use of cloud computing has increased in many organizations. Subashini and Kavitha [1] argue that small and medium companies use cloud computing services for various reasons including because this services provide fast access to their applications and reduce the infrastructure costs. With the increasing growth of data to be managed, distributed the distributed storage system provide a reliable platform for storing massive amounts of data over the network. A real life business model of distributed storage is cloud storage (e.g., Amazon S3[3] and Window Azure[5]), which enables enterprises and individuals to outsource their data backups to third-party repositories in the internet. Cloud Storage provides provides an on-demand remote backup solution. However, use a single cloud storage vendors raises concerns such as having a single point of failure [2].

We introduce system providing fault-tolerant storage over multiple cloud storage providers. users often store sensitive information with cloud storage providers but these providers may be untrusted. Dealing with “single cloud” providers is predicted to become less popular with customers due to risks of service availability failure and the possibility of malicious insiders in the single cloud. The cloud computing use has rapidly increased, security in cloud computing still

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2015

considered a major issue in the cloud computing environment. In this paper Resume maintaining system offers the service. Service based on cloud computing concept by using window SharePoint technology. System will provide the facility to store the user details with educational and work experience details. And additional work is resume file storing facility into the cloud via share point application and will allow users to access them. Here the service application provide and will allow user to upload their resume file to the cloud, via SharePoint application. And also will provide an automated facility of checking the uploaded resume file links. If the system finds that the link is broken then the automated report will be generated. Report will be generated for broken links and the application owner will see the report on his dashboard administrator. After it can request the users to upload the new file to establish the broken links back. In this system it uses SharePoint technology, SharePoint technology provide an solution platform for the professional Microsoft.NET development tools and technologies. SharePoint technologies offer a manageable and scalable server platform for hosting application on an internet. In this system, we proposed a system which facilitates to end users to upload their details in the site. We have kept the data into the cloud using SharePoint framework. We have stored the resume into the SharePoint site and mirrored them. We have the SharePoint timer job which performs the validation of the files .Job runs on hourly basis and checks for the files uploaded and for their mirror copy as well. When it encounters either of them is invalid then it is designed in such a way that it automatically makes the copy from available to non-available.

II. LITERATURE REVIEW

Yuchong Hu, Henry C. H. Chen, Patrick P. C. Lee, Yang Tang [9] in this paper, they examined and focuses on cloud failures, when a cloud fails permanently. To provide fault tolerance for cloud storage, recent studies propose to stripe data across multiple cloud vendors. However, if a cloud suffers from a permanent failure and loses all its data, then so need to repair the lost data from other surviving clouds to preserve data redundancy. They present a proxy-based system for multiple-cloud storage called NCCloud, which aims to achieve cost-effective repair for a permanent single-cloud failure. NCCloud is built on top of network-coding-based storage schemes called regenerating codes. Specifically, they propose an implementable design for the functional minimum storage regenerating code (F-MSR), which maintains the same data redundancy level and same storage requirement as in traditional erasure codes (e.g., RAID-6), but uses less repair traffic. In this paper they implement a proof-of-concept prototype of NCCloud and deploy it atop local and commercial clouds. They validate the cost effectiveness of FMSR in storage repair over RAID-6, and show that both schemes have comparable response time performance in normal cloud storage operations.

Alexandros G. Dimakis, Kannan Ramchandran, Yunnan Wu, and Changho Suh [8] in this paper they discussed on node failure when coding is used. When coding is used, the repair problem arises. If a node storing encoded information fails to create encoded information at a new node. Distributed storage systems often introduce redundancy to increase reliability. When coding is used, the repair problem arises: if a node storing encoded information fails, in order to maintain the same level of reliability we need to create encoded information at a new node. This amounts to a partial recovery of the code, whereas conventional erasure coding focuses on the complete recovery of the information from a subset of encoded packets. The consideration of the repair network traffic gives rise to new design challenges. Recently, network coding techniques have been instrumental in addressing these challenges, establishing that maintenance bandwidth can be reduced by orders of magnitude compared to standard erasure codes.

Yuchong Hu, Chiu-Man Yu, Yan Kit Li, Patrick P. C. Lee, John C. S. Lui [7] in this paper they show that a class of regenerating codes, which are based on the concept of network coding, can improve the data repair performance over traditional storage schemes such as erasure coding. However, there remain open issues regarding the feasibility of deploying regenerating codes in practical storage systems. This paper present NCFS, a distributed file system that realizes regenerating codes under real network settings. NCFS transparently stripes data across multiple storage nodes, without requiring the storage nodes to coordinate among themselves. It adopts a layered design that allows extensibility, such that different storage schemes can be readily included into NCFS. They deploy and evaluate our NCFS prototype in different real network settings. In particular, they use NCFS to conduct an empirical study of different storage schemes, including the traditional erasure codes RAID-5 and RAID-6, and a special family of regenerating codes that

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2015

are based on E-MBR [4]. Work provides a practical and extensible platform for realizing theories of regenerating codes in distributed file systems.

Henry C.H. Chen, Yuchong Hu, Patrick P.C. Lee, and Yang Tang [6] in this paper it proposed proxy-based storage system for fault-tolerant multiple-cloud storage. To provide fault tolerance for cloud storage, recent studies propose to stripe data across multiple cloud vendors. However, if a cloud suffers from a permanent failure and loses all its data, so need to repair the lost data with the help of the other surviving clouds to preserve data redundancy. They present a proxy-based storage system for fault-tolerant multiple-cloud storage called NCCloud, which achieves cost-effective repair for a permanent single-cloud failure. NCCloud is built on top of a network-coding-based storage scheme called the functional minimum-storage regenerating (FMSR) codes, which maintain the same fault tolerance and data redundancy as in traditional erasure codes (e.g., RAID-6), but use less repair traffic and, hence, incur less monetary cost due to data transfer.

III. PROPOSED ALGORITHM

Proposed Project System will have logical and physical architecture shown in below:

A. Logical Architecture

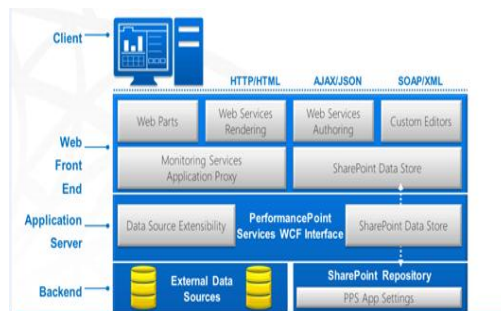


Fig.1: Logical Architecture

As shown in above diagram user will be having HTML pages view. Middle layer represents the SharePoint to design the business logic and data will be stored into SharePoint Technology as in back end.

B. Physical Architecture

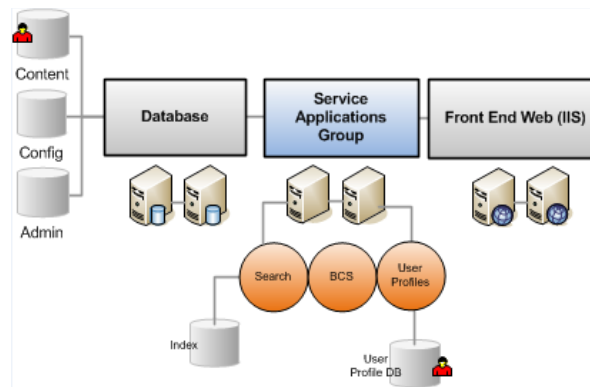


Fig.2: Physical Architecture

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2015

Above diagram shows the physical architecture of the proposed system. User request will be handled by Front End Web servers. Data will be retrieved and processed by service application group servers database server will be responsible for data handling. As mentioned in above system will be web based only hence our scope is limited for computer UI only.

Algorithms:

A. NC upload algorithm:

File upload:

1. Divide a file F into k(n-k) equal size chunks.
 $f_{i_i} \dots$ where $i=1,2,3, \dots, k(n-k)$
2. Encode k(n-k) chunks into n(n-k) chunks
 $P_{i_i} \dots$ where $i=1,2,3, \dots, n(n-k)$
Where each P_i is formed by line combination of the k(n-k) chunks.
Let, $EM = [a_{ij}]$
Where, EM is encoding matrix of
 $n(n-k) \times k(n-k)$
& $i=1,2, \dots, n(n-k)$
 $j=1,2, \dots, k(n-k)$
also,
ECV (encoding coefficient vector)- a row vector of EM which k(n-k) elements.
ECVi- denotes i th row vector of EM
3. Compute each P_i as,

$$P_i = \sum_{j=1}^{k(n-k)} a_{ij} F_j \text{ for } i = 1, 2, \dots, n(n-k)$$

4. The code chunks are then evenly stored in the n storage nodes, each having (n-k) chunks.
Also whole EM is stored in a metadata object that is then replicated to all storage nodes.

B. NC download algorithm:

1. First download ECVs using the corresponding metadata object.
2. Select k nodes out of the n storage nodes.
3. Download k(n-k) code chunks from the selected k node.
4. Then form a $k(n-k) \times k(n-k)$ square matrix from ECVs of k(n-k) code chunks.
5. The original k(n-k) native chunks,
= inverse of square matrix * code chunks
Where, inverse of square matrix exists between MDS property is maintained.

C. NC repair algorithm:

Repairs:

(For a permanent single node failure)

1. Download the encoding matrix from a surviving node
2. Select one ECV from each of the (n-1) surviving nodes i.e. $ECV_{i1}, ECV_{i2}, \dots, ECV_{i(n-1)}$
3. Generate a repair matrix
 $RM = [\gamma_{ij}]$
Where $i=1,2,3 \dots, n-k$
& $j=1,2,3 \dots, n-1$
4. Compute the ECVs for the new code chunks &

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2015

Reproduce a new encoding matrix
i.e.

$$ECV^i = \sum_{j=1}^{n-1} \gamma_{ij} ECV_{ij} \text{ for } i = 1, 2, \dots, n - k.$$

Then reproduce a new encoding matrix EM formed by substituting ECV of EM of the failed node with corresponding new ECVs.

5. Verify if each of the encoding matrices EM forms a full rank or not.
This is done by counting all $n_{c_{k \subseteq k}}$ subset of k nodes .

Where k –working nodes

6. If chunks are not available then download the actual chunk data & regenerated new chunk data.

MODIFIED ALGORITHM:

NC algorithm –upload-download-repairs:

1. Upload a file F on a given path
 f_{i_t} -file of size M
2. Make the copy of file F available on different locations
F1, F2, F3, ... Fn
Where n-number of nodes
F1-file of size M on node 1
Fn-file of size M on node n
3. Compute each P_i as,

$$P_i = \sum_{j=1}^{k(n-k)} a_{ij} F_j \text{ for } i = 1, 2, \dots, n$$

4. Where ,
k-any integer no. such that $k < n$
 a_{ij} -any element i th row & j th column of encoding matrix
 P_i - encoded element of the matrix
5. A file F of size M will be checked by automated process
6. If a file is not present then the replica of respective file will be copied to the location.
7. Then download ECVs using the corresponding metadata object.
8. Select k nodes out of the n storage nodes.
9. Download k (n-k) code chunks from the selected k node.
10. Then form a k (n-k)*k(n-k) square matrix from ECVs of k(n-k) code chunks.
11. The original k(n-k) native chunks,
= inverse of square matrix* code chunks
Where, inverse of square matrix exists between MDS property is maintained.

MATHEMATICAL MODEL:

Let,

n- total numbers of nodes
EM-encoded matrix
ECV-encoded coefficient vector
RM- repair matrix
k-Integer value s t $k < n$
 $RM = [\gamma_{ij}]$

Where,

γ_{ij} is randomly selected

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2015

$i-1,2,\dots,n-k$
 $j-1,2,\dots,n-1$

$$n_{c_k \subseteq k}$$

where k are the selected nodes which are available and working. This algorithm assign to the Cloud services. In this experiment we analyzed different algorithms and compared those algorithm.

IV. EXPERIMENTAL RESULTS

For experimental result we calculate time with increasing no. of files and examine peak period delay with respect to traffic intensity

Figures shows the results:

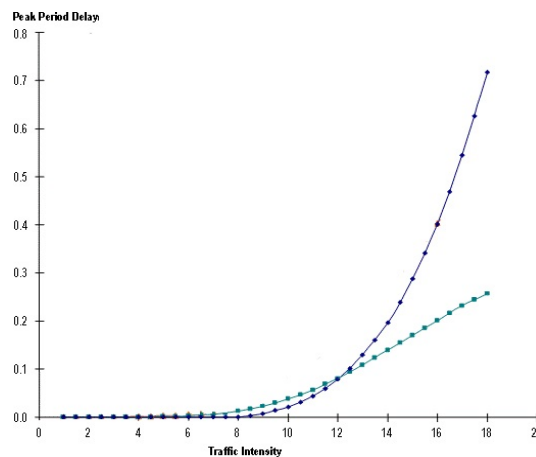


Fig. 3: Peak period delays vs traffic intensity

Above graph is plotted between peak period delays and traffic intensity. As shown in the above fig. as number of file is increasing with time.

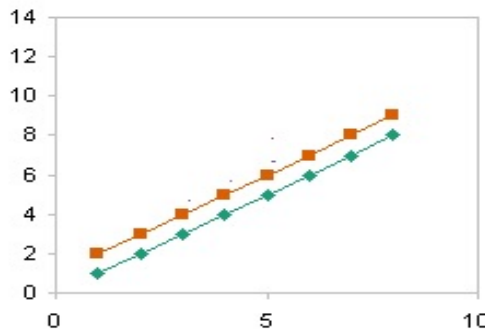


Fig. 4: Time vs no. of files

Above graph is plotted between time and no. of files. We are comparing this system with the previous system, our system is more efficient , time saving, fault tolerant.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2015

V. CONCLUSION AND FUTURE WORK

Our research suggests the Resume maintenances system is designed in such a way that it uses file repairing functionality automatically. Files are distributed at multiple locations. The files are checked through the scheduler and checked for availability. On unavailability is found then from available location the file will be replaced or copied to unavailable location and log. This system will offer a service based on cloud computing concept using SharePoint Technology which provide the best integrated server for office applications. Also SharePoint Technologies offer a manageable and scalable server platform. We introduce system providing fault-tolerant storage over multiple cloud storage providers. One key feature of the system is data reliability which generally refers to the redundancy of data storage.

In future, it can include Daily report of the monitoring activity i.e. how many times files are monitored average time required to complete the activity. Also Automated email notification can be send to the users if the file is unavailable totally. and If the file is modified by actual user then an alert functionality can be used to notify both user and administrators.

VI. ACKNOWLEDGEMENT

We would like to express our gratitude and sincere regards to the following people to whom we are grateful for their support and help. We would like to thanks Dr.Y.V.Chavan, Principal, P.V.P.I.T. for providing us with excellent facilities and valuable guidance. We would like to express my profound and sincere gratitude to Prof.N.D.Kale HOD, Computer Department who has given all cooperation and help to complete this paper. We would like to express my profound and sincere gratitude to Prof. V.S. Nandedkar, PG Coordinator, Computer Department who has given all her cooperation and help to complete this paper.

REFERENCES

- [1] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing", Journal of Network and Computer Applications, 34(1), pp 1-11,2011
- [2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A View of Cloud Computing. Communications of the ACM,53(4):50–58, 2010.
- [3] Amazon S3. <http://aws.amazon.com/s3>, 2010
- [4] K. V. Rashmi, N. B. Shah, P. V. Kumar, and K. Ramchandran. Explicit construction of optimal exact regenerating codes for distributed storage. In Proc. of Allerton Conference, 2009
- [5] Windows Azure. <http://msdn.microsoft.com/en-us/windowsazure/default.aspx>, 2010
- [6] Henry C. H. Chen, Yuchong Hu, Patrick P.C.Lee, and Yang Tang. NCCloud: Applying Network Coding for the Storage System in a Cloud-of-Clouds. IEEE Trans. On computers,vol-63,no-1,January 2014
- [7] Y. Hu, C.-M. Yu, Y.-K. Li, P.P.C. Lee, and J.C.S. Lui, "NCFS: On the Practicality and Extensibility of a Network-Coding-Based Distributed File System," Proc. Int'l Symp. Network Coding (NetCod '11), 2011
- [8] A.G. Dimakis, P.B. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network Coding for Distributed Storage Systems," IEEE Trans. Information Theory, vol. 56, no. 9, pp. 4539-4551, Sept. 2010
- [9] Y. Hu, H. Chen, P. Lee, and Y. Tang. NCCloud: Applying Network Coding for the Storage Repair in a Cloud-of-Clouds. In Proc. Of USENIX FAST, 2012
- [10] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung. Network information flow. IEEE Trans. on Information Theory, 46(4):1204–1216, Jul 2000
- [11] J.S. Plank, "A Tutorial on Reed-Solomon Coding for Fault-Tolerance in RAID-Like Systems," Software Practice & Experience, vol. 27, no. 9, pp. 995-1012, Sept.1997
- [12] H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, RACS: A Case for Cloud Storage Diversity In Proc. of ACM SoCC, 2010.